Design Document of Receiver Node Software Modifications for CSC Group 1

Created by Pam Beard
Created on August 18, 1998

1   Purpose

The purpose of this modification to the Station Controller Computer (SCC) software is to convert it into Receiver Node Software. The baseline for the Receiver Node Software will be the CSSAR version of SCC.

The instruments, which will be controlled by the Node software, are Microdyne 1200MRC Receivers, Microdyne 1620PC Combiners and a HP E1301 RF Matrix Switch.

2   Reference Document

▪ CSC Group 1 Requirements Document (rev 2.0)

3   Requirements

### 3.1   Microdyne 1200MRC Receivers, 1620PC Combiners and HP E1301 RF Matrix Switch

The RF Matrix Switch routes the LHCP and RHCP signals from the Muticoupler of the 11m, 7.3m #1, 7.3m #2 and TOTS Systems to the bank of 10 1200-MRC Telemetry Receivers and 5 1620 Diversity Combiners. The LHCP and RHCP switches, which route data to a particular set of receivers, will be treated as one switch by the software. From the RF Matrix Switch each pair of 1200MRC Receiver will be injected with LHC and RHC data, respectively, from one the four sources. The receivers are numbered #1 LHC, #1 RHC, #2 LHC, #2 RHC, #3 LHC, #3 RHC, #4 LHC, #4 RHC and #5 LHC and #5 RHC where LHC means Left Hand Circular Polarized and RHC means Right Hand Circular Polarized. The data will flow through the receivers and be sent to the associated 1620PC Combiner. The Diversity Combiners are labeled #1, #2, #3, #4 and #5 and are each wired directly to the corresponding LHC and RHC pair of Receivers.

Upon power up or turned online, the software will read available filters from the receiver. It will use the available filters as selections on the control and status screens. When configuring, if the filter selected is not available use the next higher filter and send and flashing pink pop up window to the user interface.

Configuration function will change to allow you to turn "on" or "off" and instrument. By turning a configuration "on", the configuration for that instrument will be loaded when the configuration file is applied. Turning an instrument "off" will not load the configuration for that instrument when the configuration is applied. Instead, the instrument state will be untouched. This will need to be added to all the configuration screens.
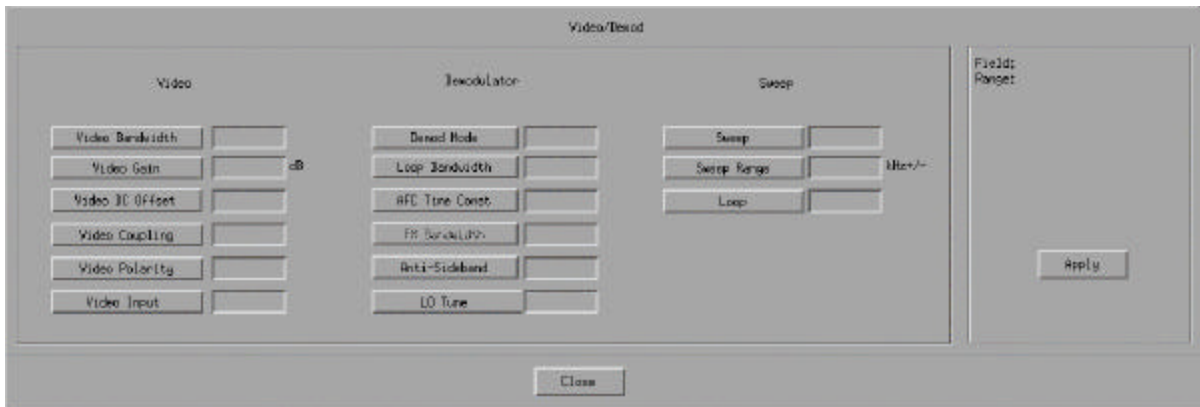
3.1.1   New user interface screens

3.1.1.1   Top level screen
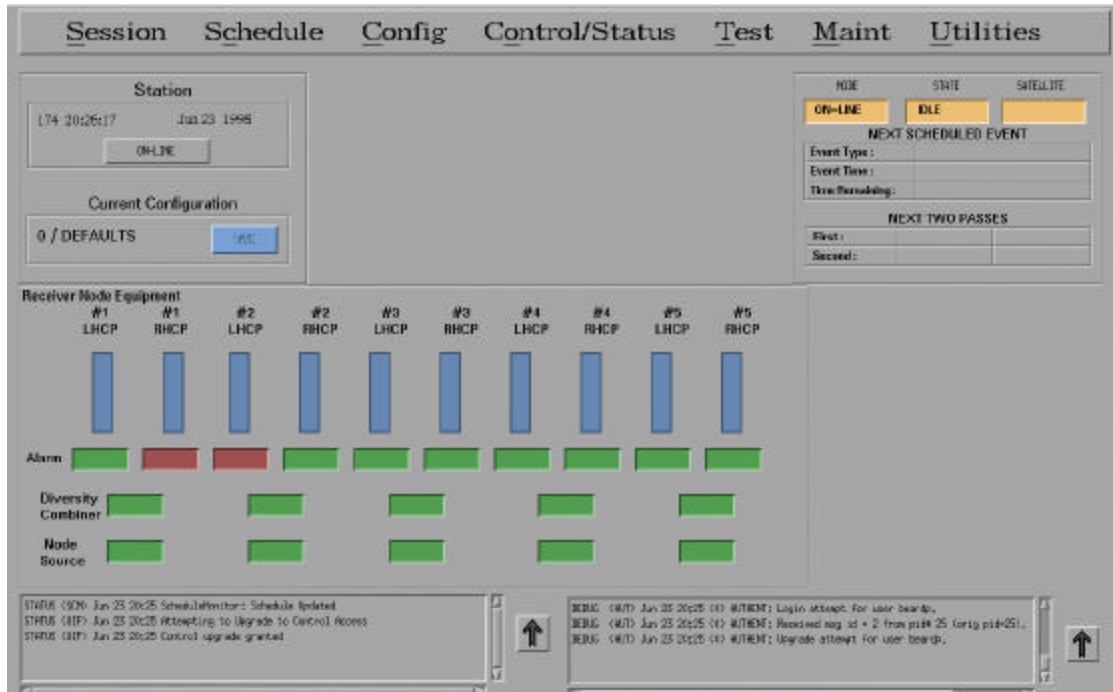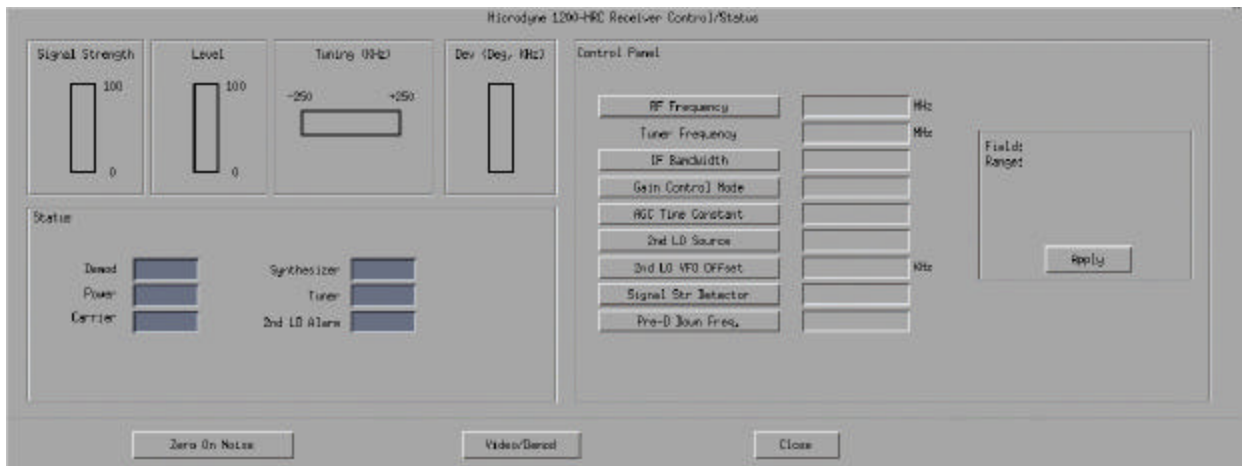

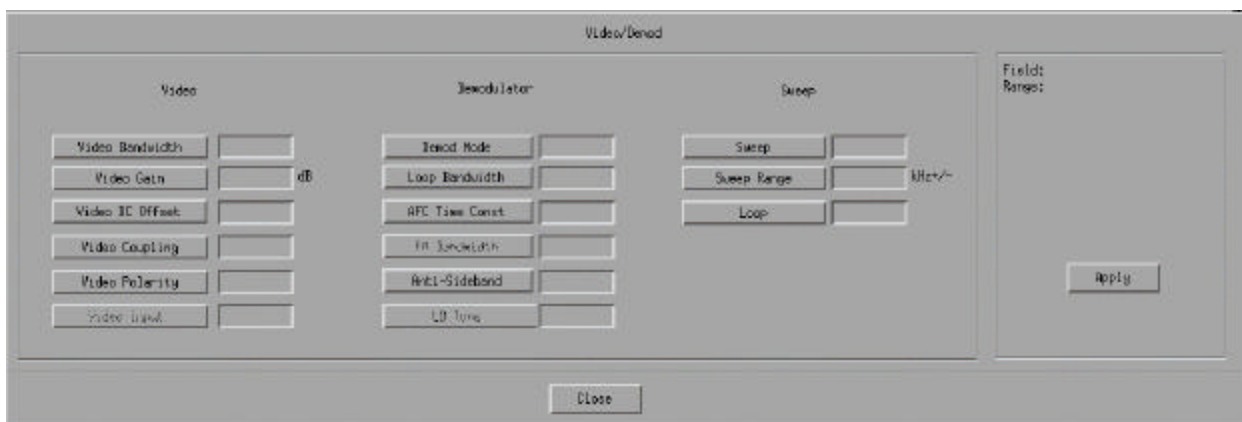#### *3.1.1.2   1620- PC Combiner Screens:*
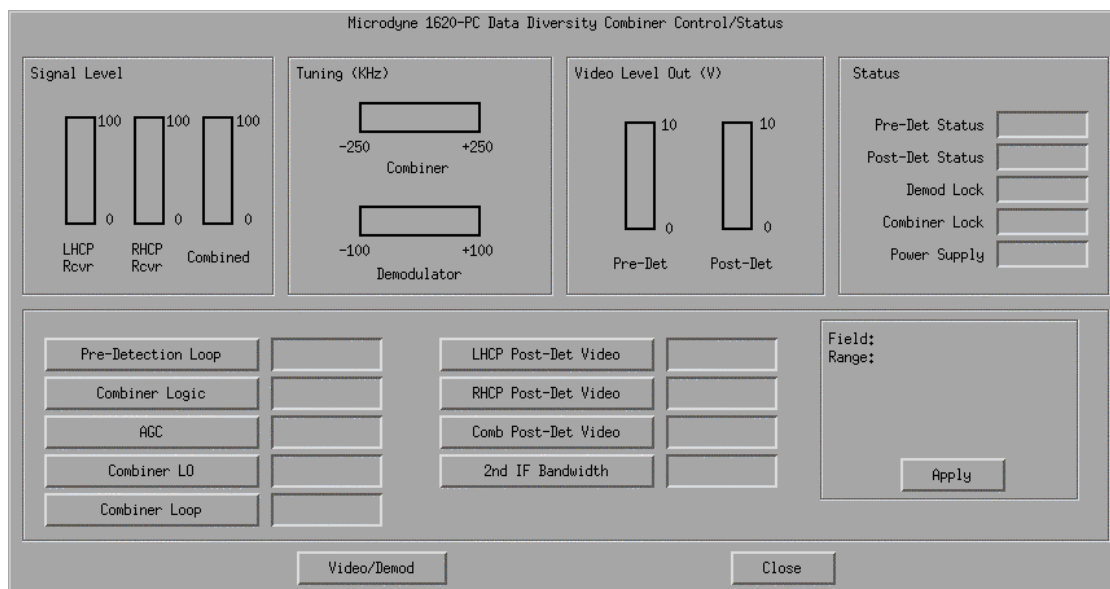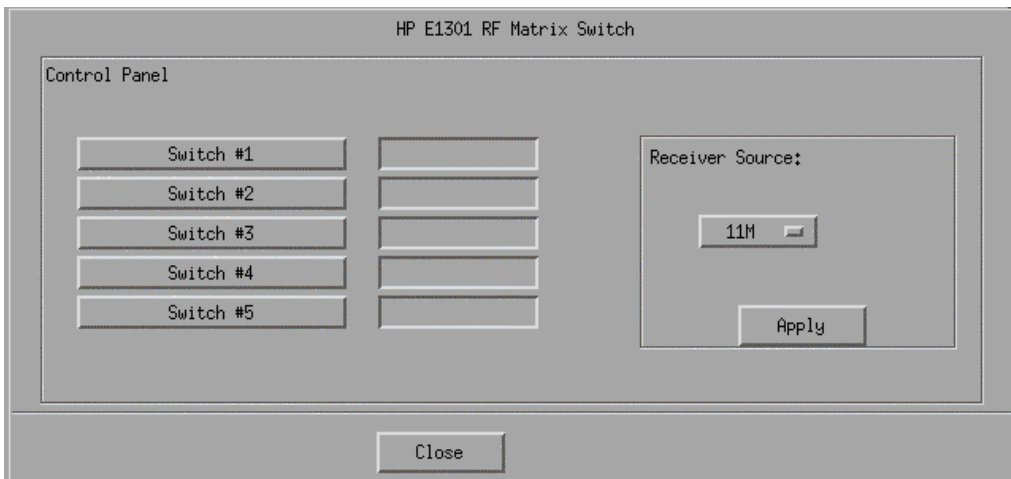

When Video/Demod is clicked:

3.1.1.3   1200-MRC screens:

When Video/Demod is clicked:

3.1.1.4   HP E1301 RF Matrix Switch

3.1.2   Files to modify:

3.1.2.1   UIF dir

Rcvr_md1200mrc.c (new) – add all screens associated with the new receiver (control/status and configuration). Include the ability to turn on and off configuration of the different instances. Each time the screen is entered, request available filter information. For status and control, allow only available filters to be selected. For configuration, indicate which filters are available, but allow selection of all filters.

Toplevel.c – Remove screen sections not needed for this project. Add additional receiver and combiner status indicators and source indications. Add the "click on an area of the screen" feature. Add the receivers, combiners and switch to the status/control menu. Add way to get to new calibrate screen. Gray out menu items that does not apply.

Uiextern.h - add prototype that need to be known outside there files.

Cmbr_md1620.c – modify to add new items for control and configuration. Add the ability to turn on and off configuration of the different instances.

Rfswitch_e1301.c (new) – all screens associated with the new RF switch receiver (control/status and configuration). Add the ability to turn on and off configuration of the different instances.

Uiprotos.h – add prototype that need to be known outside there files.

Viddem_md.c – add additional fields to video/demod screens of both receiver and combiner.

Calibrate.c – (new) add calibration screen for groups of receivers and combiner. Need to take in consideration that auto zero function takes up to 1 minute to complete.

3.1.2.2   Include dir

Project.h – define the instances of all the new instruments, add device offline mapping, configuration information, configuration default file. Modify status log instrument data.

instrume.h – add new instruments and instances to instrument list.

Probably more header files than this will be affected for things like prototypes for new functions, etc.

3.1.2.3   Statusct/cmd_rsp dir:

Offlinef.c – During power and when md1200mrc is turned online, have it poll to see what available filters it has. Also, add feature to send a pop-up window to user interface when a status or command is sent to an instrument which is offline.

3.1.2.4   Statusct/cmd_rsp dir:

Md1620pc.c – add any additional commands needed to enhance screen to meet requirements.

Md1200mrc.c (new) – create new instrument file to talk to the 1200 mrc receiver (similar commands to the md700mrc.c, this would be a good starting point). Need to include a function which polls instrument to see what available filter it has a keep track of this information. If user tries to select an unavailable filter, select the next higher filter and send a pink flashing popup window to the user interface.

Hpe1366sw.c (new) - add all new commands for switch and add the virtual instrument handling of switch as virtual instrument. (i.e. handle two switches as one).

3.1.2.5   Statusct/cmd_rsp:

Devicrsp.c – add way to handle auto zero commands to the 1200 which may take minute to complete.

3.2   Receiver Node/Master Interface

These changes are similar to UNIX/NASA code modifications under tasks 281 and 293 in Continuus.

3.2.1   Status Packet data.

The Receiver Node' s remote interface will provide status packet data about the schedule, receivers and combiners. The data will be available on the TCP/IP socket defined as rci_send (similar to the 11m system) even when a remote user is not logged in.

The status packets will be sent at a default rate of once every 10 seconds. The Remote Master Computer will have the capability to change the Status Packet rate from the default message rate of once every 10 seconds to a rate of once every 6 seconds to 25 seconds. This rate will also be the rate at which data is collected for the status log. This rate will be set with a unique Ethernet message. The rate would apply to all instruments.

 The data provided in the status packets is as follows:
- Receiver Status
  - Signal Strength (dB)
  - Receiver Lock Status (Lock/Unlock)
- Combiner Status
  - Combined Signal Strength (dB)
  - Combiner Lock Status (Lock/Unlock)
  -

The Status Log Type Packet Information will be provided at all times, within a track and outside a track.

3.2.2   Pass log data.

During a scheduled pass, status data is collected in binary files. Once the pass is complete and during postpass, the binary files are read and an ASCII summary file is produced which contains time tagged one line entries covering the status of the receivers and combiners. In the ASCII file, the locked status is represented as a 1 and the not locked status as 0. There is one log for each satellite pass taken. The naming convention is SatelliteName.day.hr.min.sec.

The status that is recorded is the same status that is reported in the status packets. The rate that the data is collected in the pass log is twice the rate at which the packet data is sent out.

```
    TIME              CL CL CL  CL CL SStr SStr SStr SStr SStr RL RL RL RL RL RL RL
RL RL RL SStr SStr  SStr SStr  SStr SStr SStr SStr SStr SStr

                      1   2  3   4  5  1    2    3    4    5    1L 1R 2L 2R 3L 3R 4L
4R 5L 5R  1L    1R    2L   2R   3L   3R   4L   4R   5L   5R   \n";

 ttttttttttttttttt  x, x, x, x, x, xxxx,xxxx,xxxx,xxxx,xxxx, x, x, x, x, x, x, x,
x, x, x,xxxx,xxxx,xxxx,xxxx,xxxx,xxxx,xxx,xxx,xxx,xxx,xxxx\n"
```

- Receiver Status
    - Signal Strength (dB)
    - Lock Status (Lock/Unlock)

- Combiner Status
    - Combined Signal Strength (dB)
    - Combiner Lock Status (Lock/Unlock)

3.2.3   Files to be modified:

3.2.3.1   Include directory:

rciipcms.h – add RCI_STATUS_LOG and RCI_STATUS_OUTPUT to last_rci_message_id and add structure for remote message.

logipcms.h – add LOG_PERIOD_CHANGE log message ids to change log period. And add the following #define STATUS_PACKET_LEN 80, #define MAX_LOG_PERIOD   25.0, and #define MIN_LOG_PERIOD 6.0.

ipcs_msgs.h - add rci_status_log_type rci_status_log to messages.

rci_pkt.h – Add NASA's packet header structures.

Log_type.h – add log structure for the new 1200 receiver.

Project.h – Add or modify structures related to status logging.

3.2.3.2   Status_l directory:

log_mes.c – change message to send status packets instead of terminal messages.

log_stat.c – Make status log accept and argument for the default status log rate. In the main loop, add cases to handle log period change.

loginst.c – Add log off all the instrument status (all the instances of receivers and combiners to match receiver node) . Also add the ability to turn off and on writing to binary files for pass log file.

logstat.c – Add function to enable and disable writing data to a binary file rather than installing and uninstalling messages. Replace updateTeminialStatus calls with build_status_packet calls for instruments in receiver node. Make variable the rate at which install messages are requested. Modify status log file to only contain data on the instruments in the receiver node. There will be no pedestal angle data, therefore start the pass log file base on the first data point it the instrument status binary file and create new pass log file entries at twice the install message rate.

loguptm.c – Change updateTerminalStatus to build_status_packet, change sendTerminalMessage to send_rci_message. Remove all messages except those related to the receiver node equipment and add additional code for handling all instances of receiver node equipment.

3.2.3.3   *Remote/client directory:*

rci_clnt.c - Modify to handle continuos status packet output; modify routine so that an error during start up will not cause message overflows. Add additional header information for NASA to make messages in same format as the 11 meter.

3.2.3.4   *Remote/server directory:*

 rcisrvr.c - Modified to handling of NASA's packet header.

3.2.4   Message Definitions

All messages are an ASCII string, which begin with a three unique letter mnemonic and end with a carriage return, line feed. Each message description below includes a representation of the message string, a description of items in the string, and its related instrument.

3.2.4.1   *Receiver Status Update (REC)*

Message string: status, "RECxx,ss,l\r\n",

        where

                xx   Receiver Number (1L : #1LHC, 1R :#1RHC)
                ss   Signal Strength (dB)
                l   Lock Status (1 = LOCK, 0 = UNLOCK)

3.2.4.2   *Combiner Status Update (COM):*

Message string: "STC,cccc\r\n",

        where

                xx   Combiner Number (1- 5)
                ss   Combined Signal Strength (dB)
                c   Combiner Lock Status (1 = LOCK, 0 = UNLOCK)

3.2.4.3   *Message Status Period (MSP)*

Message string: "MSP,ss.s\r\n"

        where

                        ss.s   seconds (6 – 25 seconds)

4 Miscellaneous

Track function in Window NT needs to look to see if ACU is offline so that is doesn't continue to send messages out to equipment which is offline.